OBSTACLE AVOIDANCE ROBOT APPLYING FUZZY CONTROL SYSTEM

LIEW CHIA WOON

A project report submitted in partial

fulfillment of the requirement for the award of the

Master of Electrical Engineering

Faculty of Electrical and Electronic Engineering

Universiti Tun Hussein Onn Malaysia

JULY, 2014

ABSTRACT

This project primarily addresses the design and implementation of fuzzy logic based controller for obstacles avoidance robot. Some techniques, sensors and controller have been applied to fulfill the requirements of the robot. The robot able to interact with an unknown environment using a reactive strategy determined by the sensors information. The obstacle avoidance controller is a two inputs and two outputs system. The inputs are two proximity measurement of the obstacles distance, and the outputs are the velocities of the two wheels. The robot is driven by two independent DC motors. The ultrasonic sensors are mounted at left, front and right side of the mobile robot. It can detect the obstacles within the range 3 cm to 300 cm. The kinematic model was developed to control the movement of the robot's wheels. The detection of the obstacles by the sensors activates the controller which simply attempts to avoid the robot with the obstacles. Once the robot avoids the obstacles, the robot will keep sense for the new environment. Based on these signals, the controller control the velocity of left and right wheels thus making the robot to move forward and turning at the same time, i.e., the robot avoid the obstacle by turn left with left motor velocity is 6.57 cm/s, right motor velocity is 8.29 cm/s and angular velocity is 0.066154 rad/s when the obstacle detected from left sensor is 100 cm, right sensor is 100 cm and front sensor is 50 cm. Different obstacles distance orientation also been tested and the robot response is working as expected.

ABSTRAK

Projek ini untuk menyampaikan reka bentuk dan pelaksanaan pengawal *fuzzy logic* untuk mengelak halangan robot. Beberapa teknik, pengesan dan pengawal telah digunakan untuk memenuhi keperluan robot ini. Robot ini boleh berinteraksi dengan persekitaran yang tidak diketahui dengan menggunakan strategi reaktif yang ditentukan oleh pengesan. Kawalan *fuzzy logic* sebagai kawalan pintar digunakan untuk robot bagi mengelakkan halangan-halangan. Pengesan ultrasonik digunakan untuk mengesan halangan-halangan dan sebagai pemboleh ubah bagi masukan pengawal. Ia boleh mengesan halangan-halangan dalam lingkungan 3 cm ke 300 cm. Model kinematik telah dibangunkan untuk mengawal pergerakan bagi roda robot. Pengawal mengawal pergerakan roda kiri, roda kanan dan sudut pusingan dalam masa yang sama berdasarkan isyarat dari masukan pengesan, iaitu robot mengelakkan halangan dengan pusing ke kiri dengan halaju motor kiri 6.57 cm/s halaju motor kanan 8.29 cm/s dan halaju sudut 0.066154 rad/s apabila halangan dikesan daripada pengesan kiri adalah 100 cm, pengesan kanan 100 cm dan pengesan depan 50 cm. Halangan dengan orientasi jarak yang berbeza telah diuji dan tindak balas robot berfungsi seperti yang dijangkakan.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

cm	-	Centimetre
DC	-	Direct Current
FIS	-	Fuzzy Inference System
FLC	-	Fuzzy Logic Controller
ICC	-	Instantaneous Center of Curvature
KHz	-	Kilohertz
L	-	distance between the two wheels
PIC	-	Peripheral Interface Controller
PID	-	Proportional-Integral-Derivative
PWM	-	Pulse Width Modulation
r	-	nominal radius of each wheel
R	-	instantaneous curvature radius of the robot trajectory, relative to the mid-point of the wheel axis.
V_l	-	Linear velocity of left wheel (cm/s)
V_r	-	Linear velocity of right wheel (cm/s)
ω_l	-	angular velocity of left wheel
ω_r	-	angular velocity of right wheel

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Project Background

In recent years, research and application employing non-analytical methods of computing such as fuzzy logic, evolutionary computation, and neural networks have demonstrated the utility and potential of these paradigms for intelligent control of complex system. In particular, fuzzy logic has proven to be a convenient tool for handling real world uncertainty and knowledge representation [1, 2-4].

Fuzzy logic based approach has been successfully to control nonlinearity, uncertainty and complexity system recently. Fuzzy control system is suited to apply for autonomous mobile robots which have complex control architectures. The design of the autonomous robots is complex because of the uncertain input signal from the unknown environments and sensor inputs. The autonomous robot cannot be developed by using only the microcontroller and input signals from the ultrasonic sensors. Thus, the behaviour-based control architecture such as obstacle avoidance behaviour and wall following behaviour will be used to control the robot.

The purpose of this study is to propose and develop an obstacle avoidance robot that using fuzzy control system. Ultrasonic range finders or ultrasonic sensors are used to avoid collision with unexpected obstacles. The significant contribution expected through this study is an optimal autonomous control system that will automatically avoid the unknown objects.

1.2 Problems Statement

There are many approaches have been proposed to solve the problem, such as PID controller, behaviour-based methods, neural network methods, and some others methods [5,6]. Problem with the differential term in PID controller is that small amounts of noise can cause large amounts change in the output. In the previous research, performance analysis of the conventional PID controller and fuzzy logic controller has been done by the use of Matlab and Simulink and in the end comparison of various time domain parameters is done to prove that the fuzzy logic controller has better stability, small overshoot and fast response as compared to PID controller. Another problem faced with PID controllers is that they are linear. The performance of PID controllers in non-linear systems is variable. Often PID controllers are enhanced through methods such as scheduling or fuzzy logic.

Behaviour-based methods have been widely used for navigation of mobile robot such as line navigation and wall navigation method. Using wall navigation, not all the paths for robot movement is along the walls. Robot will detect far distance or false reading of wall if the robot crossing the space located at walls. By using line navigation, not all surface of the floor suitable can be planted a line as guidance.

Furthermore, in the sensors part, most sensors will occasionally generate noise in their output. For example, an infrared sensor might indicate the infrared light is present when actually no light is present. Or, a proximity sensor might give a questionable reading. If the noise is predictable enough, it can be filtered out in software. The noisy IR sensor might not be trusted until it gives some number of consecutive readings in agreement with one another. Affiliated with the problem of noisy data is missed data, where for either electrical or software reasons, a sensor reading is not detected or a touch sensor jams and fails to trigger. Besides, sensor data can also be adversely affected by ambient environmental conditions or battery strength.

The reasons of choosing fuzzy logic controller because it can work with less precise inputs, it doesn't need fast processors, and it is more robust than other non-linear controllers.

1.3 Project Objectives

The objectives for this project are:

a) To develop a two wheels obstacle avoidance robot that can avoid unknown or unexpected obstacles in an unknown environment.
b) To design Sugeno fuzzy logic controller for the obstacle avoidance robot.
c) To embedded the Sugeno fuzzy control into the robot.
d) To test the robot performance at real time.

1.4 Project Scopes

In order to achieve the objectives of the project, there are several scopes had been outlined. The scopes of the project are:

a) Develop the hardware for the system.
b) Ultrasonic sensors are applied as an obstacle detection of the robot. The detection range is within 3 cm until 300 cm.
c) Robot develop consists of two wheels and each of them are drive using DC motor.
d) Fuzzy logic inference method used in this project is Sugeno Fuzzy Logic.
e) The robot able to avoid the obstacles on the flat surface, from any starting point.

1.5 Thesis Outline

The thesis is divided into five chapters. Chapter 1 the introduction, Chapter 2 is devoted to a survey of the literature on obstacles avoidance mobile robot. Chapter 3 discusses the methodology of the project which is deals with the analysis of a proposed fuzzy logic technique with three types of membership functions to avoid the obstacles in an unknown environment. Besides that, the components use to develop the hardware of the robot also be discussed. Chapter 4 deals with how the fuzzy logic controller taking the decision to avoid the obstacles and the simulation result of Matlab Simulink. This chapter also consists of experimental setup, behaviour of the robot during avoid obstacles and simulation results. Chapter 5 conclude of the project and discuss ideas for further work as suggested.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In order to design and develop an obstacles avoidance robot by using fuzzy logic controller, extensive research on the fuzzy controller need to be fulfilled. This section will discuss previous studies that have been accomplished by other researchers in the same area.

2.2 Description of Previous Methods

Some previous methods that have been developed by other researchers are reviewed.

Qing-yong*et all.* [7] have developed a bahavior-based mobile robot navigation in unknown environment by using fuzzy logic. There are four basic behaviors used in their mobile bot navigation which is goal behaviour, obstacle avoidance behaviour, tracking behaviour, and deadlock disarming behaviour. These basic behaviors were implemented by using fuzzy logic controller. A behaviour controller is designed to determine the control action of the mobile robot. The simulation experimental results showed that the proposed architecture enables the mobile robot to safely achieve the goal without colliding.

Limen, *et all.* [8] developed a new fuzzy intelligent obstacle avoidance control strategy for wheeled mobile robot, which is composed of two fuzzy logic controllers and an intelligent coordinator. Fuzzy obstacle controller will generate obstacle avoidance commands according to the target orientation information and obstacle information when the robot detecting obstacles via its onboard sensors.

Intelligent coordinator is designed to coordinate run-to-goal fuzzy controller and fuzzy obstacle and generate ultimate robot control commands.

Mobile robot control using type-2 fuzzy logic system was developed by Pisit and Supachai [9]. Type-2 fuzzy sets are described by membership functions that are themselves fuzzy and output processor of a type-2 fuzzy contains two components: type-reduction and defuzzification. The type-2 fuzzy logic controller will process data output to control the direction of the mobile robot movement. The behaviour-based control was obstacle avoidance and corridor following. The results obtained demonstrate the efficiency of type-2 fuzzy logic control system and the ability to solve the problems.

Ability of a robot to avoid collision with unforeseen or dynamic obstacles while it is moving towards a target or tracking a path is a vital task in autonomous navigation. Navigation strategies can be classified to global path planning and local path planning.

In global path planning, information about the obstacles and a global model of environment is available which mostly Configuration space, Road map, Voronoi diagram and Potential field techniques are used to plan obstacle-free path towards a target. However, in real world a reliable map of obstacle, accurate model of environment and precise sensory data is unavailable due to uncertainties of the environment. While the computed path may remain valid but to response the unforeseen or dynamic obstacles, it is necessary for the robot to alter its path online. In such situations, Fuzzy logic can provide robust and reliable methodologies dealing with the imprecise input with low computational complexity [10]. Different obstacle avoidance approaches were developed during past decades which proposed effective solution to the navigation problems in unknown and dynamic environments.

Zavlangas*et all.* [11] developed a reactive navigation method for omnidirectional mobile robots using fuzzy logic. The fuzzy rule-base generates actuating command to get collision free motions in dynamic environment. The fuzzy logic also provides an adjustable transparent system by a set of learning rules or manually. Seraji and Howard developed a behavior-based navigation method on challenging terrain using fuzzy logic. The navigation strategy is comprised of three behaviors. Local obstacle avoidance behaviour is consists of a set of fuzzy logic rule statements which generates the robot’s speed based on obstacle distance [12].

Chee *et all.* [13] presented a two-layer fuzzy inference system in which the first layer fuses the sensor readings. The left and right clearances of the robot were found as outputs of the first-layered fuzzy system. The outputs of the first layer together with the goal direction are used as the inputs of the second-layer. Eventually, the final outputsof the controller are the linear velocity and the turning rate of the robot. The second-stage fuzzy inference system employs the collision avoiding, obstacle following and goal tracking behaviours to achieve robust navigation in unknown environments. Dadios and Maravillas proposed and implemented a fuzzy control approach for cooperative soccer micro robots. A planner generates a path to the destination and fuzzy logic control the robot's heading direction to avoid obstacles and other robots while the dynamic position of obstacles, ball and robots are considered [14].

Parhi described a control system comprises a fuzzy logic controller and a Petri Net for multi robot navigation. The Fuzzy rules steer the robot accordingto obstacles distribution or targets position. Since the obstacle's position is not known precisely, to avoid obstacles in a cluttered environment fuzzy logic is a proper technique for this task. Combination of the fuzzy logic controller and a set of collision prevention rules implemented as a Petri Net model embedded in the controller of a mobile robot enable it to avoid obstacles that include other mobile robots [15].

A fuzzy controller designed by Lilly for obstacle avoidance of an autonomous vehicle using negative fuzzy rules. The negative fuzzy rules define a set of actions to be avoided to direct the vehicle to a target in presence of obstacles [16].

2.3 Fuzzy Control System

The present project uses a Fuzzy Decision Making Controller which is a type of fuzzy logic controller (FLC). This type of Fuzzy Logic can be used for controlling a process i.e., a plant in control engineering terminology which is non-linear. The advantage of FLC is that it enables control engineers to easily implement control strategies which can be used by a human operator.

The components of FLC are an inference engine and a set of linguistic IF-THEN rules that encode the behaviour of the mobile robot. However, the main difficulty in designing a fuzzy logic controller is the efficient formulation of the fuzzy IF-Then rules. If it is easy to produce the antecedent parts of a fuzzy rule base, it is however very difficult to produce the consequent parts without expert knowledge [17]. Figure 2.1 shows the Fuzzy Decision Making controller is made up of three steps:

1) Fuzzification: Converts controller inputs into information that the inference mechanism can be easily use to activate and apply rules.
2) Rule base: A set of IF-Then rules which contains a fuzzy logic quantification of the expert's linguistic description of how to achieve good control.
3) Defuzzification: This converts the conclusions of the interface mechanism into actual inputs for the process.

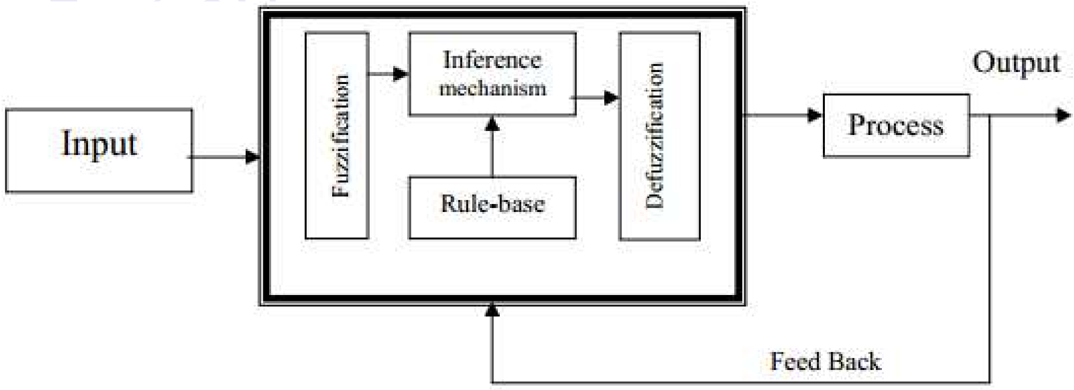

Figure 2.1: Fuzzy Logic Controller [18].

There are three main fuzzy inference systems (fuzzy logic approximators) which is Mamdani, Sugeno, and Tsukamoto type. In this project, Sugeno system is used.

Sugeno, or Takagi-Sugeno-Kang, method of fuzzy inference. Introduced in 1985,it is similar to the Mamdani method in many respects. The first two parts of the fuzzy inference process, fuzzifying the inputs and applying the fuzzy operator, are exactly the same. The main difference between Mamdani and Sugeno output membership functions are either linear or constant.

A typical rule in a Sugeno fuzzy model has the form

If input 1 = x and input 2 = y, then output is z = ax + bx + c

For a zero-order Sugeno model, the output level z is a constant (a=b=0).

The ouput level z_i of each rule is weighted by the firing strength w_i of the rule.

For example, for an AND rule with input 1 = x and input 2 = y, the firing strength is

$w_i = AndMethod\ (F_1(x), F_2(y))$

Where $F_{1,2}(.)$ are the membership functions for input 1 and 2. The final output of the system is the weighted average of all rule outputs, computed as

$$\text{Final Output}= \frac{\sum_{i=1}^{N} w_i z_i}{\sum_{i=1}^{N} w_i} \tag{2.1}$$

A Sugeno rule operators as shown in the following diagram.

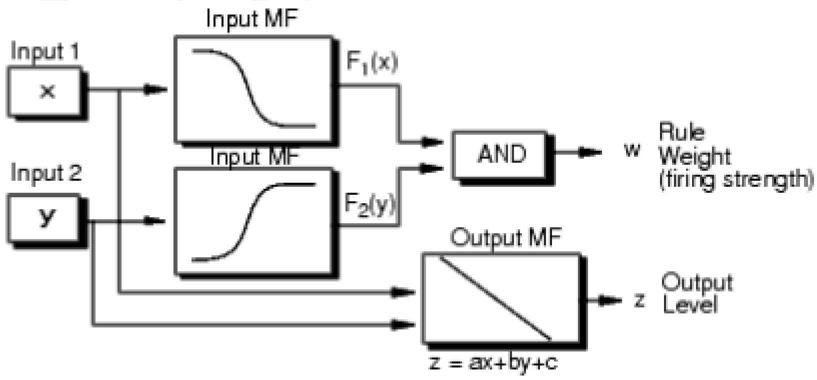

Figure 2.2: Sugeno rule operator.

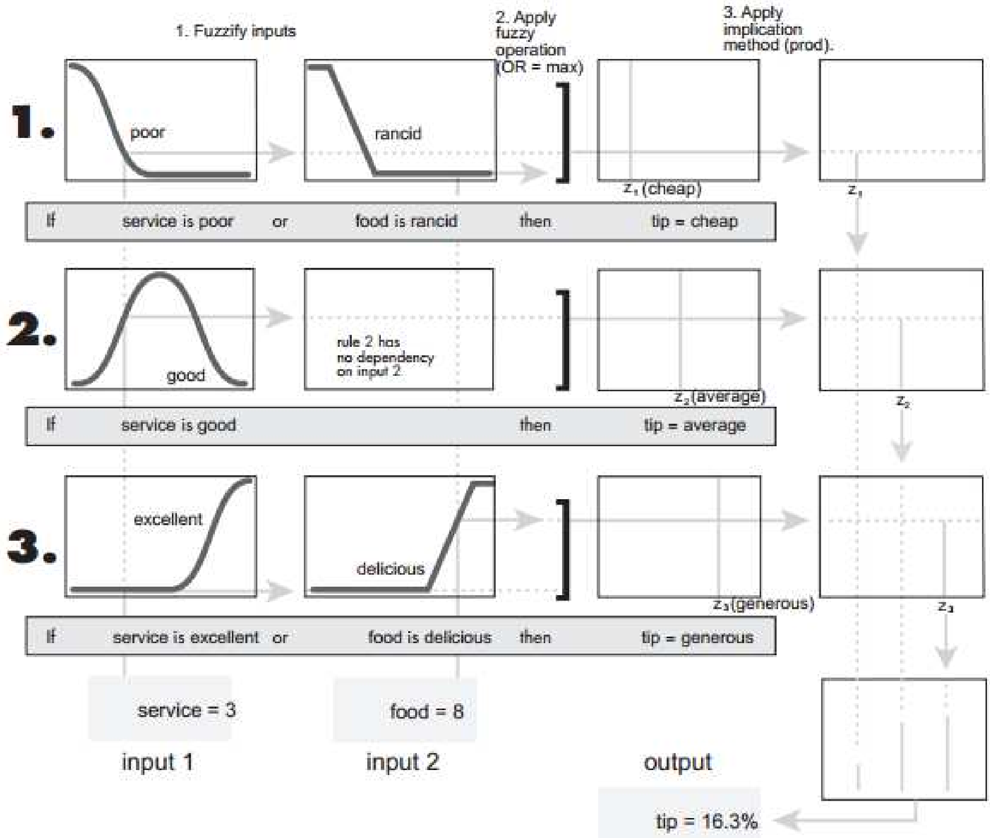

Figure 2.3: The example of fuzzy tipping model developed by using Sugeno system.

Fuzzy based systems are constructed so that generated outputs change in a smooth and continuous manner, regardless of inputs crossing set boundaries. System inputs undergo three transformations to become system outputs. First a fuzzification process uses predefined membership functions to map each system input into one or more degrees of membership. Then the rules in the rule base are evaluated by combining degrees of membership to form output strengths. Lastly the defuzzification process computes system outputs based on these output strengths and membership functions [19].

2.4 Kinematics Model

The robot has two identical parallel, wheels attached to both sides of the vehicle which are controlled by two independent DC motors. The velocity of the center of mass of the robot is orthogonal to the wheels' axis, L. The center of mass of the robot is located in the middle of the axis connecting the wheels (L). The Figure of kinematics model is shown in Figure 2.2.

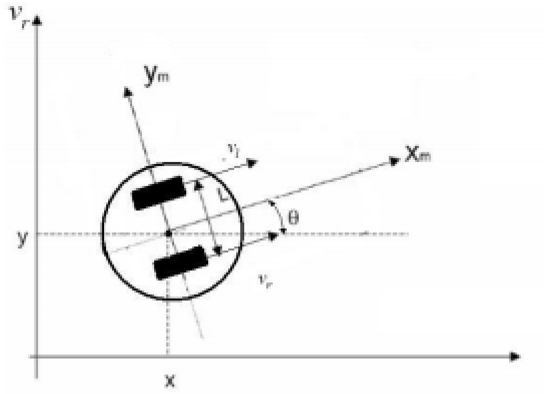

Figure 2.4: Kinematics model of the robot.

The angular velocities (ω_l and ω_r) of the two wheels are independently controlled [20]. The equations that are used to build a Matlab Simulink model of the are given by relation [21]:

$V_r(t) =$ linear velocity of right wheel

$V_l(t) =$ linear velocity of left wheel

ω_r(t)=angular velocity of right wheel

ω_l(t) = angular velocity of left wheel

$r =$ nominal radius of each wheel

$L =$ distance between the two wheels

R= instantaneous curvature radius of the robot trajectory, relative to the mid-point of the wheel axis.

ICC = Instantaneous Center of Curvature

$$R - \left(\frac{L}{2}\right) = \text{Curvature radius of trajectory decribed by left wheel}$$

$$R + \left(\frac{L}{2}\right) = \text{Curvature radius of trajectory decribed by right wheel}$$

With respect to ICC the angular velocity of the robot is given as follows:

$$\omega(t)=\frac{V_r(t)}{R+(\frac{L}{2})} \tag{2.2}$$

$$\omega(t)=\frac{V_l(t)}{R-(\frac{L}{2})} \tag{2.3}$$

$$\omega(t)=\frac{V_r(t)-V_l(t)}{L} \tag{2.4}$$

The instantaneous curvature radius of the robot trajectory relative to the mid-point axis is given as

$$R=\frac{L\,(V_l(t)+V_r(t))}{2(V_l(t)-V_r(t))} \tag{2.5}$$

Therefore, the linear velocity of the robot is given as

$$V(t)=\omega(t)R=\frac{V_r(t)+V_l(t)}{2} \tag{2.6}$$

The kinematics equations in the world frame can be represented as follows.

$$X(t) = V(t)Cos\theta(t) \tag{2.7}$$

$$Y(t) = V(t)Sin\theta(t) \tag{2.8}$$

$$\theta(t) = \omega(t) \tag{2.9}$$

This implies

$$X(t) = \int_0^t V(t)\cos\big(\theta(t)\big)\,dt \tag{2.10}$$

$$Y(t) = \int_0^t V(t)\sin\big(\theta(t)\big)dt \tag{2.11}$$

$$\theta(t) = \int_0^t \omega(t)dt \tag{2.12}$$

The above equation can also be represented in the following form

$$\begin{bmatrix} V_x(t) \\ V_y(t) \\ \theta(t) \end{bmatrix} = \begin{bmatrix} cos\theta & 0 \\ sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V(t) \\ \omega(t) \end{bmatrix}$$

$$= \begin{bmatrix} V(t)cos\theta \\ V(t)\sin\theta \\ \omega(t) \end{bmatrix}$$

$$= \begin{bmatrix} ((V_r + V_l)cos\theta)/2 \\ ((V_r + V_l)sin\theta)/2 \\ (V_r - V_l)/2 \end{bmatrix} \tag{2.13}$$

These are the equations that are used to build a model of the robot. These equations were used to simulate the robot in MATLAB Simulink. The fuzzy logic controller was tested and fine-tuned on this model.

2.5 Typical Mobility Configurations

The accuracy of odometry measurements for dead reckoning is to a great extent a direct function of the kinematic design of a vehicle. Because of this close relation between kinematic design and positioning accuracy, one must consider the kinematic design closely before attempting to improve dead reckoning accuracy.

2.5.1 Differential Drive

Figure 2.5 shows a typical differential drive mobile robot. In this design incremental encoders are mounted onto the two drive motors to count the wheel revolutions. The robot can perform dead reckoning by using simple geometric equations to compute the momentary position of the vehicle relative to a known starting position.

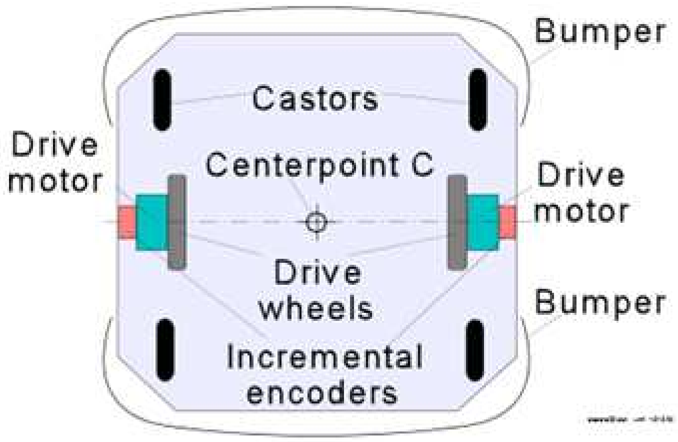

Figure 2.5: A typical differential-drive mobile robot (robot view) [22].

For completeness, rewrite the well-known equations for odometry below (also, see [Klarer, 1988; Crowley and Reignier, 1992]). Suppose that at sampling interval I the left and right wheel encoders show a pulse increment of $N_l$and N_r, respectively.

Suppose further that

$$c_{m\,=}\; nD_n/nC_e \tag{2.14}$$

Where

C_m = conversion factor that translates encoder pulses into linear wheel displacement

D_n = nominal wheel diameter (in mm)

c_m = encoder resolution (in pulse per revolution)

n = gear ratio of the reduction gear between the motor (where the encoder is attached) and the drive wheel.

The incremental travel distance for the left and right wheel, $\Delta U_{l,i}$ and $\Delta U_{r,i}$ according to

$$\Delta U_{l/r,i} \;=\; c_m N_{l/r,i} \tag{2.15}$$

and the incremental linear displacement of the robot's center point C, denoted $\Delta U_{l,i}$ according to

$$\Delta U_i = (\Delta U_r + \Delta U_l)/2 \tag{2.16}$$

Next, the robot's incremental change of orientation

$$\Delta\theta_i = (\Delta U_r - \Delta U_l)/L \tag{2.17}$$

where L is the wheelbase of the vehicle, ideally measured as the distance between the two contact points between the wheels and the floor.

The robot's new relative orientation θ_i can be computed from

$$\theta_i = \theta_{i-1} + \Delta\theta_i \tag{2.18}$$

and relative position of the center point is

$$x_i = x_{i-1} + \Delta U_i \cos\theta_i \tag{2.19a}$$

$$y_i = y_{i-1} + \Delta U_i sin\theta_i \tag{2.19b}$$

Where

$x_i y_i =$ relative position of the robot's center point c at the instant i [22].

CHAPTER 3

METHODOLOGY

3.1 Project Methodology

This chapter deals specifically with the implementation of the fuzzy logic controller in the microcontroller as well as in Matlab. Simulink was used to simulate the mobile robot for testing as well as tuning the fuzzy logic controller. Firstly, the fuzzy logic toolbox that was available was used. The fuzzy logic controller was built using the fuzzy logic toolbox, which contains graphical interface.

Sugeno method inference is used because Sugeno method is computationally efficient which suitable for microcontroller application. There was different types of membership functions were tested and also the ranges of the membership functions can be tweaked to search for an optimum result. The rules developed also can be changed and the effect observed and defuzzification was tested.

3.2 Fuzzy Controller Design and Modeling

In this project, the Fuzzy Logic Controller (FLC) designed to control the motion of the robot. There have two inputs which is distance sensor and distance difference of the robot. The two outputs are left motor velocity (*left motor-v*) and right motor velocity (*right motor-v*). Thus, the FLC is a two inputs and two outputs system.

In this project, Matlab's Fuzzy Logic Toolbox was used to design the FLC. The Matlab's Fussy Logic Toolbox contains functions, graphical user interfaces (GUI) and data structures that allow the user to quickly design, test, simulate and modify a fuzzy inference system.

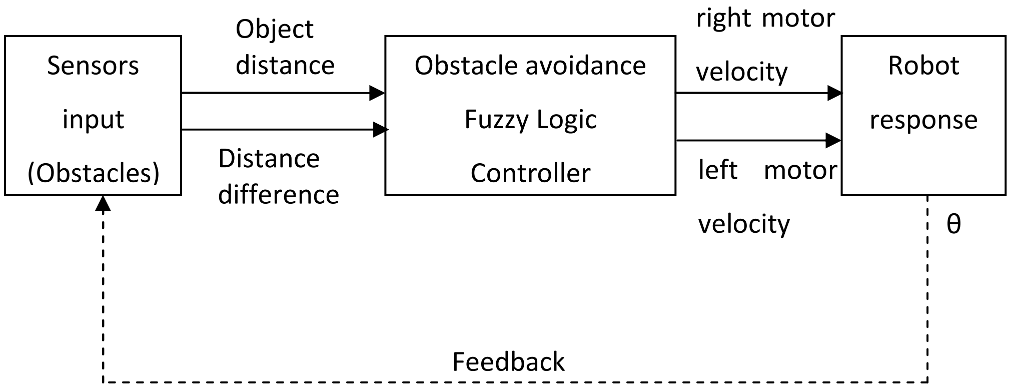

Figure 3.1: Block diagram of fuzzy behavior robot control architecture

3.2.1 Obstacle Avoidance

Distance crisp value between robot and surrounding objects measure by ultrasonic sensor circuit for distance measurement which used to build the fuzzy membership function. The acquired information from the sensors shows that there exist obstacles nearby robot. When a robot is close to an obstacle, it must change its velocity and steering angle to avoid the obstacle.

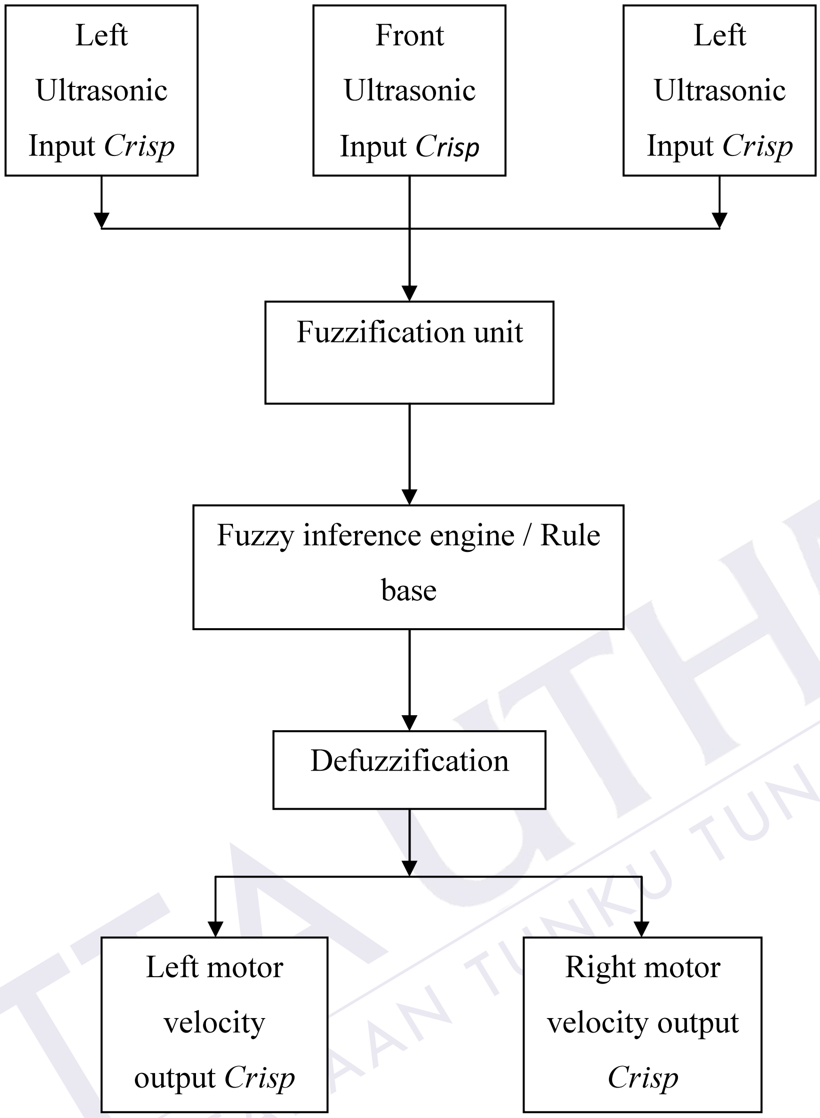

Figure 3.2: Block diagram of fuzzy based obstacle avoidance robot control behavior architecture.

The fuzzy rules used for obstacle avoidance by the robots are listed in Table 3.3 as rules 1 to 9. All the rules in the table are obtained heuristically.

Table 3.1: Fuzzy rule for velocity of the left motor.

Left motor velocity, (cm/s)		**Object distance, (cm)**		
		Near	Medium	Far
Object difference, (cm)	Negative	Fast	Medium	Medium
	Zero	Slow	Medium	Fast
	Positive	Fast	Slow	Slow

Table 3.2: Fuzzy rule for velocity of the right motor.

Right motor velocity, (cm/s)		**Object distance, (cm)**		
		Near	Medium	Far
Object difference, (cm)	Negative	Slow	Slow	Slow
	Zero	Slow	Medium	Fast
	Positive	Slow	Medium	Medium

Table 3.3: List of rules for obstacle avoidance.

Rules No.	Object distance	Object difference	Left motor	Right motor
1.	Near	Negative	Fast	Slow
2.	Near	Zero	Medium	Slow
3.	Near	Positive	Fast	Slow
4.	Medium	Negative	Medium	Slow
5.	Medium	Zero	Medium	Slow
6.	Medium	Positive	Slow	Medium
7.	Far	Negative	Medium	Slow
8.	Far	Zero	Medium	Slow
9.	Far	Positive	Slow	Medium

3.2.2 Robot Behavior

The robot has two wheels powered by separate DC motors and two castors are provided for stability of the robot. The sensors for measuring the distances around it from front obstacle distance, left obstacle distance, and right obstacle distance. The distance between the robots and obstacles act as repulsive forces for avoiding the obstacles.

According to the information acquired by the robot using their sensors, some of the fuzzy control rules are activated. The outputs of the activated rules are combined by fuzzy logic operations to control the velocities and steering angle of the driving wheels of the robot. These are denoted by left motor and right motor, for the velocity of the left wheel and the velocity of the right wheel of the robot respectively.

In this project, three types of membership functions are considered. First one is the three-membership function having two trapezoidal members and one triangular member. Linguistic variables such as "Near", "Medium" and "Far" are taken for three-membership function.

Some of the fuzzy control rules are activated according to the information acquired by the robot using sensors. The outputs of the activated rules are weighted by fuzzy reasoning and the velocities of the driving wheels of the robot are calculated. Left wheel velocity and right wheel velocity are denoted as left motor and right motor respectively as shown in Table 3.3.

Linguistic variables "Slow"; "Medium" and "Fast" are defined for left motor velocity and right motor velocity for three membership functions. Terms like "Slow", "Medium", and "Fast" are considered for left wheel velocity and right wheel velocity for three membership functions.

The parameter input and output membership functions were defined as Table 3.4 until Table3.7 respectively. Values in the tables present values of x-axis in a kind of singular points. That is, " 0 : 0 : 60 : 150" is values of x-axis in singular points for a fuzzy membership function "Near" of input fuzzy variables and "6" is value of x-axis in singular points for a fuzzy membership function "Slow" of output fuzzy variables.

Table 3.4:Parameters for object distance

Term	**Range of Membershi Function (cm)**	**Membership Function**
Near	0 : 0 : 60 : 150	Trapezoidal
Medium	90: 145 : 200	Triangular
Far	150: 220 :300 : 300	Trapezoidal

Table 3.5: Parameters for object difference

Term	**Range of Membershi Function (meter)**	**Membership Function**
Negative	-126 .-70 . -14	Trapezoidal
Zero	-56 .0 .56	Triangular
Positive	14 .70 .126	Trapezoidal

Table 3.6: Parameters for left motor

Term	**Range (velocity)**
Slow	6
Medium	8
Fast	10

Table 3.7: Parameters for right motor

Term	**Range (velocity)**
Slow	6
Medium	8
Fast	10

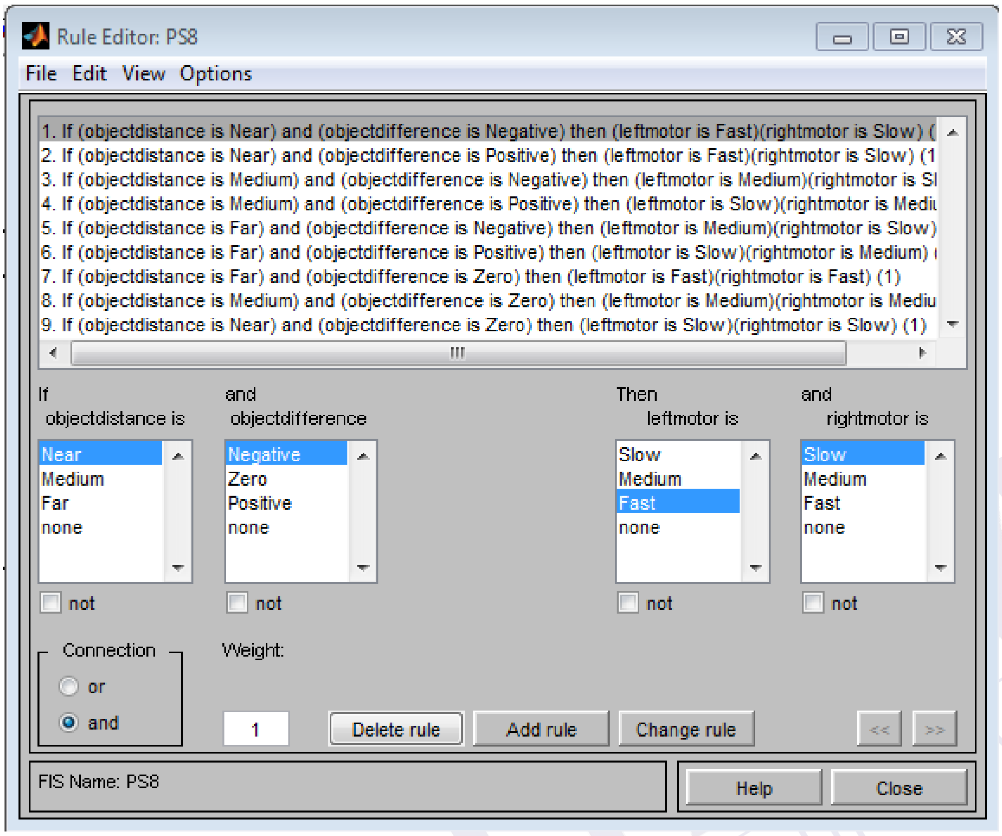

Figure 3.3: Rules viewer for the membership functions.

These parameters used to generate different fuzzy rules**:**

If *(object distance is near and object difference is negative* Then *(left motor is fast and right motor is slow).*

If *(object distance is near and object difference is zero* Then *(left motor is slow and right motor is slow).*

If *(object distance is near and object difference is positive* Then *(left motor is fast and right motor is slow).*

If *(object distance is medium and object difference is negative* Then *(left motor is medium and right motor is slow).*

If *(object distance is medium and object difference is zero* Then *(left motor is medium and right motor is medium).*

If *(object distance is medium and object difference is positive* Then *(left motor is slow and right motor is medium).*

If *(object distance is far and object difference is negative* Then *(left motor is medium and right motor is slow).*

If *(object distance is far and object difference is zero* Then*(left motor is fast and right motor is fast).*

If *(object distance is far and object difference is positive* Then*(left motor is slow and right motor is medium).*

By fuzzy reasoning and the defuzzification, rules related to the obstacle avoidance behaviours are weighted to determine an appropriate control action, i.e., the velocities, left motor and right motor of the robot's wheels as shown in Figure 3.3, the values of the parameters are decided empirically.

3.3 Operation of the Robot

General program for the robot operation is created for robot rules. Figure3.4 shows the flow chart of the general robot operation. At first step the program read all sensors value that attached to the robot. Then the distance value of obstacle sensor is calculated. The program receives ranging data from the sensors then uses this to conduct the fuzzy logic operations. Robot then will make the turning to avoid the obstacle depend on the output of the fuzzy logic program. When the turning process is complete, the sensor detects or read again for the next input. If there is obstacle again, the avoiding process still running until there is no obstacle.

If no obstacle is detected, the robot will move forward in fast condition. After each movement, then the sensor is read again to continue the next movement process. From the sensor reading, the mathematical rule is applied to find the obstacle distance and obstacle difference value.

REFERENCES

1. R. Malhotra and A. Sarkar. Development of a Fuzzy Logic Based Mobile Robot for Dynamic Obstacle Avoidance and Goal Acquisition in an Unstructured Environment., *IEEE/ASME Proceeding of International Conference on Advanced Intelligent Mechatronics.* 20- 24 July 2003. Kobe. pp. 235-247.
2. D. Ratner and P. McKerrow (2003). Navigation an Outdoor Robot along Continuous Landmarks with Ultrasonic Sensing. *Robotics and Autonomous Systems,* 45(1). Retrieved November 23, 2013, from doi: 10.1016/S0921-8890(03)00096-4
3. R. Carelli and E. O. Freire (2003). Navigation Outdoor and Wall-Following Stable Control for Sonar-Based Mobile Robots. *Robotics and Autonomous Systems,* 45(3). Retrieved November 24, 2013, from doi: 10.1016/j.robot.2003.09.005
4. R. Benporad, M. Di Marco and A. Tesi. Wall-Following Controller for Sonar-Based Mobile Robots. *Proceeding of IEEE Conference on Decision and Control*, San Diego. 10-12 December 1997. San Diego. pp. 3063-3068.
5. W. L. Xu and S. K. Tso. Real-time Self-reaction of a Mobile Robot in Unstructured Enviroments using Fuzzy Reasoning. *Engng Applic Aritif. Intell.* 1996. pp. 475-485.
6. Anmin Zhu and Simon X.Yang. A Fuzzy-logic Approach to Reactive Navigation of Behavior-based Mobile Robots. *IEEE International Conference on Robotics & Automation*. April, 2004, pp. 5054-5050.
7. BAO Qing-yong, LI Shun-ming, SHANG Wei-yan, & AN Mu-jin (2009). A Fuzzy Behavior-Based Architecture for Mobile Robot Navigation in Unknown Enviroments. *IEEE*. pp. 257-261.
8. Limin Ren, Weidong Wang & Zhijiang Du (2012). New Fuzzy Intelligent Obstacle Avoidance Control Strategy for Wheeled Mobile Robot. *IEEE*. 1732-1737.

9. PisitPhokharatkul & SupachaiPhaiboon. Mobile Robot Control Using Type-2 Fuzzy Logic System. *IEEE.* 2004. 296-299.

10. Yanik, P., Ford, G., & Howell, B. An introduction and literature review of fuzzy logic applications for robot motion planning. *Proceedings of ASEE Southeast Section Conference*. Western Carolina University. 2010. 1-10.

11. Zavlangas, P.G., Tzafestas, S.G., & Althoefer, K. Fuzzy Obstacle Avoidance and Navigation for Omnidirectional Mobile Robots. *ESIT.* 2000. 375-382.

12. Seraji, H. & Howard, A. Behavior-based robot navigation on challenging terrain: A fuzzy logic approach. *IEEE Trans. Robotics and Automation.* 2002. (18): 308-321.

13. Chee, B. Y., Lang, S.Y.T., & Tse, P.W.T. Fuzzy mobile robot navigation and sensor integration. *IEEE International Conference on Fuzzy Systems.* New Orleans, LA. 1996. 1: 7-12.

14. Dadios, E. & Maravillas, O. Cooperative Mobile Robots with Obstacle and Collision Avoidance Using FuzzyLogic. *Proceedings of IEEE International Symposium on Intelligent Control.* October 27-30, 2002. Vancouver, Canada: IEEE, 2002. 75-80.

15. Parhi, D.R. Navigation of mobile robots using a fuzzy logic controller. *Journal of intelligent and robotic systems.* 2005. 42(3): 253–273.

16. Lilly, J.H. Evolution of a negative-rule fuzzy obstacle avoidance controller for an autonomous vehicle. *IEEE Trans. Fuzzy System.* 2007. 15: 718-728.

17. J. M. Lucas, H. Martinez-Barbera, & F. Jimenez. Multi-objective Evolutionary Fuzzy Modeling for the Docking Maneuver of an Automated Guided Vehicle. *IEEE International Conference on System, Man and Cybernetic.* 2005. 4: pp. 2999-3004.

18. Rerngwut Choomuang and Nitin Afzulpurkar. Hybrid Kalman Filter/Fuzzy Logic based Pasition Control of Autonomous Mobile Robot. *International Journal of Advanced Robotic Systems.* 2005. 2: pp. 197-208.

19. The MathWorks, Inc (2009), MATLAB Fuzzy Logic Toolbox.

20. Vamsi Mohan Peri. *Fuzzy Logic Controller for an autonomous Mobile Robot.* Master's Thesis. University of Jawaharlal Nehru Technological India; 2002.

21. Razif Rashid, I. Elamvazuthi, MumtajBegam, & M. Arrofiq. Fuzzy-based Navigation and Contol of Non-Holonomic Mobile Robot. *Journal of Computing.* 2010. 2(3): 130-137.

22. J. Borenstein, H. R. Everett, & L. Feng. *Where am I? Sensors and Methods for Mobile Robot Positioning.* United States: University of Michigan. pp. 19-20; 1996.

23. Ibrahim, M. T., Dirman Hanafi & Ruzlaini Ghoni. Autonomous Navigation for a Dynamical Hexapod Robot using Fuzzy Logic Controller. *International Conference on Modeling Optimisation and Computing, Procedia Engineering 38*. Elsevier Ltd. 2012. pp. 330-341.

.